

A new class of high order semi-Lagrangian schemes for rarefied gas dynamics.

Giacomo Dimarco*, Cory Hauck[†], Raphaël Loubère[‡]

March 16, 2016

Abstract

In this paper we generalize the fast semi-Lagrangian scheme developed in [J. Comput. Phys., Vol. 255, 2013, pp 680-698] to the case of high order reconstructions of the distribution function. The original first order accurate semi-Lagrangian scheme is supplemented with polynomial reconstructions of the distribution function and of the collisional operator leading to an effective high order accurate numerical scheme for all regimes, from extremely rarefied gas to highly collisional situation. The main idea relies on updating at each time step the extreme points of the distribution function for each velocity of the lattice instead of updating the solution in the cell centers, these extremes points being located at different positions for any fixed velocity of the lattice. The result is a class of scheme which permits to preserve the structure of the solution over very long times compared to existing schemes from the literature. We propose a proof of concept of this new approach along with numerical tests and comparisons with classical numerical methods.

Keywords: Semi-Lagrangian schemes, kinetic equations, high-order schemes, discrete velocity models, rarefied gas dynamics.

Contents

1	Introduction	2
2	Kinetic equations and Fast Kinetic Scheme	3
2.1	The kinetic model	3
2.2	The first order Fast Kinetic Scheme	4
3	High order polynomial reconstruction for the FKS scheme	7
3.1	General context for high accuracy extension	7
3.2	Linear polynomial reconstruction case	10

*Department of Mathematics and Computer Science University of Ferrara, Ferrara, Italy. & Université de Toulouse; UPS, INSA, UT1, UTM; CNRS, UMR 5219; Institut de Mathématiques de Toulouse; F-31062 Toulouse, France. (giacomo.dimarco@unife.it)

[†]Computer Science and Mathematics Division; Oak Ridge National Laboratory, US. (hauckc@ornl.gov)

[‡]Université de Toulouse; UPS, INSA, UT1, UTM; CNRS, UMR 5219; Institut de Mathématiques de Toulouse; F-31062 Toulouse, France. (raphael.loubere@math.univ-toulouse.fr)

4	Numerical experiments	12
4.1	Convergence test problem	13
4.2	Riemann problem using Sod like initial data	14
4.3	Highly oscillating kinetic problem	15
4.3.1	Mesh convergence study	17
4.3.2	Cost and efficiency study	19
5	Conclusion and perspectives	24

1 Introduction

Kinetic models are nowadays a very important tool which is used to describe many different phenomena ranging from classical fluid mechanics, plasmas and rarefied gases [13, 5, 2] to biology and socio-economy models [35]. The advantage of kinetic equations with respect to more classical approaches based on fluid mechanics equations is the possibility to take into account a larger spectrum of regimes and, consequently, of phenomena. Typically fluid mechanics describes problems which are in the so-called thermodynamics equilibrium state while the kinetic approach permits to consider both thermodynamics equilibrium as well as far from equilibrium states for the particles constituting the system under analysis.

Unfortunately, the strong capability in terms of modelization are mitigated by several numerical problems of different nature [13]. One typically would like to preserve the physical conserved quantities also at the numerical level because they characterize the steady states and, in addition, to be able to deal with different spatial and temporal scales which are often encountered in practical applications. A more fundamental problem related to kinetic numerical simulation is represented by a light version of 'the curse of dimensionality' of these models. In general the distribution function which defines the state of the system and which gives the probability for a particle to be at a given position with a given speed at a fixed time depends on seven independent variables: $(x, v, t) \in \mathbb{R}^7$. On the contrary fluid mechanics unknowns live in a \mathbb{R}^4 space: three physical space components plus the time. This makes numerical simulations of realistic multi dimensional problems demanding in terms of computational time and memory consumption.

Historically, there exists two different approaches which are typically used to tackle kinetic equations: deterministic numerical methods such as finite volume, semi-Lagrangian and spectral schemes [13] and probabilistic numerical methods such as Direct Simulation Monte Carlo (DSMC) [1, 4]. Both methodologies have strengths and weaknesses. While the first could normally reach high order of accuracy, the second are often faster, especially for solving steady problems but, typically, they exhibit lower convergence rate and difficulties in describing non stationary and slow motion flows.

For this reason, many different research works have been dedicated in the recent past to reduce some of the disadvantages of these classes of methods. In particular, concerning the drawback of the excessive computational cost for deterministic methods we recall the fast spectral approaches developed in [20, 33, 34], while for tackling the multiple scales problem we remind the work in [28, 14, 15] and the references therein to have a broad view on the problematic and the associated numerical techniques.

The recent review on deterministic numerical methods for kinetic equation [13] gives a panorama of the state of the art of numerical discretization of kinetic equation. Concerning the issues related to the excessive noise of Monte Carlo methods we quote [4] for an overview on efficient and low variance Monte Carlo methods while for applications of variance reduction techniques to kinetic equation we remind to the recent works [26, 27] and [16, 17, 10].

In this work, we specifically tackle the issue related to the construction of efficient numerical methods for the linear transport part of a kinetic model. In details, we consider the development of a new class of semi-Lagrangian methods with focus on the efficiency of the method and on the possibility to extend the method developed in [18] to arbitrarily high order polynomial reconstructions of the distribution function. Thus, the class of methods described in this paper is the natural extension of the fast kinetic scheme (FKS) described in [18, 19] which was based on the passage from the continuous kinetic model to discrete velocity models (DVM) [3, 31, 32, 30] and on a semi Lagrangian approach [6, 7, 8, 9, 24, 25, 21, 36] to solve the transport part. The FKS approach was shown to be an efficient way to solve linear transport equations which has permitted to simulate the full six dimensions on a single processor machine. Unfortunately the solutions computed with this method were limited to a first order in space and time precision. This approach was, in fact, based on a piece-wise constant reconstruction of the distribution function for each velocity of the lattice. In this paper, contrarily, we consider arbitrary polynomial reconstructions which permit to get a theoretical arbitrary high order accurate method if coupled with suitable high order splitting techniques [37, 11] for solving the full problem given by the transport plus the collision operator. The main idea is to use a semi-Lagrangian technique to compute the solution of the transport equation and then to compute the solution of the collision operator on the maxima and the minima of the distribution function instead of computing it on the grid points. This approach, as will be shown later, allows to diminish the numerical diffusion with respect to classical semi-Lagrangian methods employing the same polynomial reconstruction. Moreover this approach is well suited to preserve the structure of the solution for very long times compared to existing schemes.

The class of methods proposed can be used for discretizing different kinetic equations with different type of collision operators as it will be clarified later. Since the focus is on the way in which the transport part is solved in the numerical test section a simple relaxation type collision operator, i.e the BGK (Bhatnagar-Gross-Krook) operator [23], has been chosen. Extensions to the full Boltzmann operator and to multidimensional settings is a work in progress.

The article is organized as follows. In the first section 2, we present the model and the Fast Kinetic numerical Scheme (FKS). In section 3 we generalize the FKS to arbitrary high order reconstructions. A full example is described in the case of BGK kernel and piece-wise linear reconstructions. Several test problem results are presented in section 4 to assess the efficiency of this new high accurate numerical method exposed in this paper by comparison with classical schemes. Conclusion and future developments are finally drawn in section 5.

2 Kinetic equations and Fast Kinetic Scheme

In this section we briefly present the model and the numerical method designed in [18]. However we refer the reader to the cited papers for an exhaustive description.

2.1 The kinetic model

In the kinetic theory of rarefied gases, the non-negative function $f = f(x, v, t)$ characterizes the state of the system and it defines the density of particles having velocity $v \in \mathbb{R}^3$ in position $x \in \mathbb{R}^3$ at time $t \in \mathbb{R}^+$. The time evolution of the system is obtained through the equation

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f = Q(f). \quad (1)$$

The operator $Q(f)$, on the right hand side in equation (1), describes the effects of particle interactions and its form depends on the details of the microscopic dynamic. Independently on the

type of microscopic interactions considered, typically the operator characterizes the conservation properties of the physical system

$$\int_{\mathbb{R}^3} Q(f)\phi(v) dv = 0, \quad (2)$$

where $\phi(v) = (1, v, |v|^2)$ are commonly called the collision invariants. We denote by $U(x, t) = \int_{\mathbb{R}^3} f(x, v, t)\phi(v) dv \in \mathbb{R}^5$ the first three moments of the distribution function f . Integrating (1) against $\phi(v)$ yields a system of macroscopic conservation laws

$$\frac{\partial}{\partial t} \int_{\mathbb{R}^3} f\phi(v) dv + \int_{\mathbb{R}^3} v \cdot \nabla_x f \phi(v) dv = 0. \quad (3)$$

The above moment system is not closed since the second term involves higher order moments of the distribution function f . However, using the additional property of the operator $Q(f)$ that the functions belonging to its kernel satisfy

$$Q(f) = 0 \quad \text{iff} \quad f = M[f], \quad (4)$$

where the Maxwellian distributions $M[f] = M[f](x, v, t)$ can be expressed in terms of the set of moments $U(x, t)$, one can get a closed system by replacing f with $M[f]$ in (3). This corresponds to the set of compressible Euler equations which can be written as

$$\frac{\partial U}{\partial t} + \nabla_x \cdot F(U) = 0, \quad (5)$$

with $F(U) = \int_{\mathbb{R}^3} M[f]v\phi(v) dv$. Note that the simplest operator satisfying (2) and (4) is the linear relaxation operator [23]

$$Q_{BGK}(f) = \nu(M[f] - f), \quad (6)$$

where $\nu = \nu(x, t) > 0$ defines the so called collision frequency. On the other hand, the classical Boltzmann operator reads

$$Q_B(f) = \int_{\mathbb{R}^3} \int_{\mathbb{S}^2} B(|v - v_*|, \omega) (f(v')f(v'_*) - f(v)f(v_*)) dv_* d\omega, \quad (7)$$

where ω is a vector of the unitary sphere $\mathbb{S}^2 \subset \mathbb{R}^3$. The post collisional velocities (v', v'_*) are given by the relations

$$v' = \frac{1}{2}(v + v_* + |q|\omega), \quad v'_* = \frac{1}{2}(v + v_* - |q|\omega), \quad (8)$$

where $q = v - v_*$ is the relative velocity, the kernel B characterizes the details of the binary interactions. In this work, the construction of the numerical scheme is first done in the general case of a collision operator which only satisfies conditions (2) and (4). For the numerical simulations we consider the BGK operator which has been shown to be able to model with sufficient accuracy many different rarefied regimes [5].

2.2 The first order Fast Kinetic Scheme

The Fast Kinetic Scheme (FKS) [18] belongs to the family of so-called semi-Lagrangian schemes [8, 9, 21] which are typically applied to a Discrete Velocity Model (DVM) [3, 30] approximation of the original kinetic equation. We briefly recall the basics of the scheme in the one dimensional setting and we remind to [18] for details.

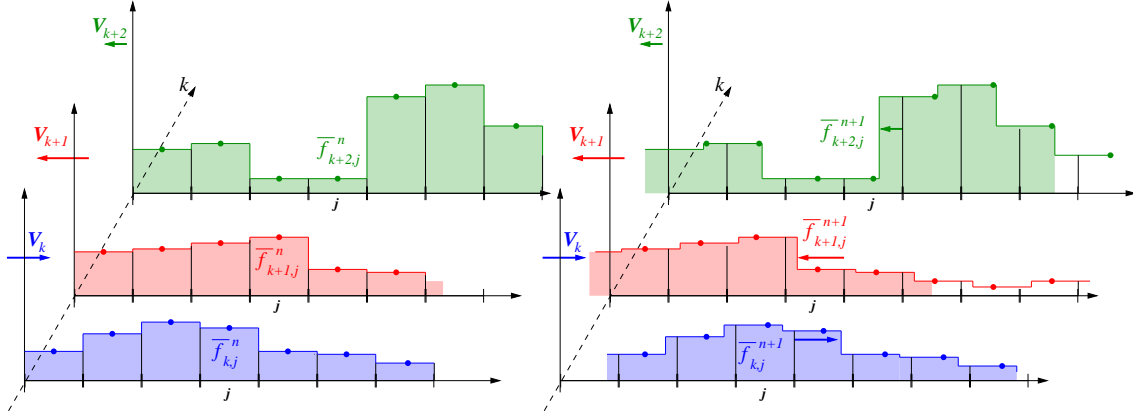


Figure 1: Illustration of the transport scheme for the first order FKS scheme. Left panel before transport step, right panel after transport step. Each discrete velocity (index k) drives its own transport equation with velocity v_k . The representation of f is made by means of a piece-wise constant function. The shape of the entire function has not changed during the transport but the cell-centered values (bullets) may have.

Let us truncate the velocity space fixing some given bounds and set a grid in velocity space of N points with Δv the grid step. The continuous distribution function f is then replaced by a vector whose components are assumed to be approximations of the distribution function f at locations v_k :

$$f_k(x, t) \approx f(x, v_k, t). \quad (9)$$

The discrete velocity kinetic model consists then of a set of N evolution equations in the velocity space for f_k , $1 \leq k \leq N$, of the form

$$\partial_t f_k + v_k \cdot \nabla_x f_k = Q(f_k), \quad (10)$$

where $Q(f_k)$ is a suitable approximation of the collision operator $Q(f)$ at location k . Observe that, due to the truncation of the velocity space and to the finite number of points with which f is discretized, the moments of the discrete distribution function f_k are such that

$$\tilde{U}(x, t) = \sum_k \phi_k f_k(x, t) \Delta v \neq U(x, t), \quad (11)$$

with $\phi_k = (1, v_k, v_k^2)$ the discrete collision invariants, are no longer the ones given by the continuous distribution f . This problem concerns all numerical methods based on the discrete velocity models and different strategies can be adopted to restore the correct macroscopic physical quantities [30, 31, 32, 22]. In order to solve this problem we adopt a L_2 projection technique for the discretized distribution which permits to project the discretized f_k in the space of the distributions for which the moments are exactly the continuous macroscopic quantities we aim to preserve. We do not enter into more details and we refer the reader to [22] for a description of this technique.

Let us now introduce a Cartesian uniform grid in physical space made of M points with Δx a scalar which represents the grid step in the physical space. Further we define a time discretization $t^{n+1} = t^n + \Delta t$ starting at t^0 , where Δt is the time step defined by an opportune CFL condition discussed next. We restrict ourselves to uniform meshes in space. Investigations are in progress to extend the present method to more complex geometries.

Each equation of system (10) is solved by a time splitting procedure. We recall here a first order splitting approach: first a transport step exactly solves the left-hand side, whereas a collision stage solves the right-hand side using the solution from the transport step as initial data:

$$\text{Transport stage} \longrightarrow \partial_t f_k + v_k \cdot \nabla_x f_k = 0, \quad (12)$$

$$\text{Collisions stage} \longrightarrow \partial_t f_k = Q(f_k). \quad (13)$$

Transport step. Let $f_{j,k}^0$ be the point-wise values at time t^0 of the distribution f , $f_{j,k}^0 = f(x_j, v_k, t^0)$. The idea behind the fast kinetic scheme is to solve the transport stage (12) continuously in space, see Fig. 1 for a sketch. To this aim we define at the initial time the function $\bar{f}_k^0(x)$ as a piece-wise continuous function for all $x \in \Omega_j$, where spacial cell $\Omega_j = [x_{j-1/2}; x_{j+1/2}]$ and $\Omega = \bigcup_j \Omega_j$. Hence starting from data $\bar{f}_{k,j}^0$ at time index 0, the exact solution of (12) is simply

$$\bar{f}_k^{*,1} = \bar{f}_k^0(x - v_k \Delta t), \quad \forall x \in \Omega. \quad (14)$$

In other words, the entire function \bar{f}_k^0 is advected with velocity v_k during Δt unit of time and the * superscript indicates that only the transport step has been solved so far. The extension of this procedure to the generic time step n gives

$$\bar{f}_k^{*,n+1} = \bar{f}_k^n(x - v_k \Delta t), \quad \forall x \in \Omega, \quad (15)$$

where now, the key observation is that the discontinuities of the piece-wise function $\bar{f}_k^n(x)$ do not lie on the interfaces of two different cells. Instead, the positions of the discontinuities depend entirely on the previous advection step and thus they may be located anywhere in the physical space. This means that if only the linear transport equation has to be solved, this approach gives the exact solution to the equation if the initial data is effectively a piece-wise constant function initially centered on the spacial mesh.

Collision step. The effect of the collisional step is to change the amplitude of $\bar{f}_k(x)$. The idea is to solve the collision operator locally on the grid points and successively extend these computed values to the full domain Ω . Thus we need to solve the following ordinary differential equation

$$\partial_t f_{j,k} = Q(f_{j,k}), \quad (16)$$

for all velocities of the lattice $k = 1, \dots, N$ and grid points $j = 1, \dots, M$. The initial data for solving this equation is furnished by the result of the transport step obtained by (15) at points x_j of the mesh at time $t^{n+1} = t^n + \Delta t$, i.e. $\bar{f}_k^{*,n+1}(x_j)$, for all $k = 1, \dots, N$, and $j = 1, \dots, M$. Then, the solution of (16), locally on the grid points, reads if, for simplicity, a forward Euler scheme is used

$$f_{j,k}^{n+1} = f_{j,k}^{*,n+1} + \Delta t Q(f_{j,k}^{*,n+1}), \quad (17)$$

where $f_{j,k}^{*,n+1} = \bar{f}_k^{*,n+1}(x_j)$. Of course many different time integrators can be employed for solving this equation. In particular special care is needed in the case in which the equation becomes stiff, refer to [14, 15] for alternative strategies. Since the time integration of the collision term is not the issue considered in this paper, we considered the simplest possible scheme, the FKS technique remains the same when other time integrators are employed. Equation (17) furnishes a new value for the distribution f at time t^{n+1} only in the cell centers of the spacial cells for each velocity v_k . However, one needs also the value of the distribution f in all points of the domain in order to

perform the transport step at the next time step. Therefore, we define a new piece-wise constant function \overline{Q}_k for each velocity of the lattice v_k as

$$\overline{Q}_k^{n+1}(x) = Q(f_{j,k}^{*,n+1}), \quad \forall x \text{ such that } \overline{f}_k^{*,n+1}(x) = \overline{f}_k^{*,n+1}(x_j). \quad (18)$$

Said differently we make the fundamental assumption that the discontinuities of Q coincide with the ones of f . Thanks to the above choice one can rewrite the collision step in term of spatially reconstructed functions as

$$\overline{f}_k^{n+1}(x) = \overline{f}_k(x, t^n + \Delta t) = \overline{f}_k^{*,n+1}(x) + \Delta t \overline{Q}_k^{n+1}(x). \quad (19)$$

This ends one time step of the FKS scheme.

Concerning the transport part of the scheme, the time step Δt is constrained by a CFL condition of type

$$\Delta t \max_k \left(\frac{|v_k|}{\Delta x} \right) \leq 1 = \text{CFL}. \quad (20)$$

while the time step constraint for the collision step depends on the choice of the operator Q . Since in the numerical test section we used a BGK operator, the time step constraint for this part has been chosen as $\Delta t \leq \nu \rho$. As observed in [18] the transport scheme is stable for every choice of the time step, being the solution for a given fixed reconstruction performed exactly. Nonetheless the full scheme being based on a time splitting technique, the error is of the order of Δt in the case of first order splitting or of order $(\Delta t)^q$ for a splitting of order q . This suggests to take the usual CFL condition in order to maintain the time splitting error small enough.

To conclude this section let us observe that time accuracy can be increased by high order time splitting methods, while spacial accuracy can be increased close to the fluid limit to a nominally second-order accurate scheme by the use of piece-wise linear reconstructions of state variables, see the details in [19]. In the next section, we will introduce a procedure which allows to increase the spacial accuracy by using high order polynomial reconstructions.

3 High order polynomial reconstruction for the FKS scheme

3.1 General context for high accuracy extension

The purpose of this section is to provide design principles to generalize the family of FKS schemes to higher accuracy in space. The FKS method employs simple piece-wise constant functions while our aim is to take into account arbitrary polynomial reconstruction of the distribution f . The goal is to obtain a class of highly accurate in space schemes and to design a method which will be viable in 3D×3D [12], relatively inexpensive and independent of the collisional operator employed: BGK, ES-BGK, Boltzmann, etc. In this section we first present the general framework and then we introduce three possible choices which permit to design different schemes. We conclude with a critical discussion which leads naturally to choose only one of the listed possibilities for the numerical experiments of the numerical section 4. While, in the first part, the method is presented in the general context valid for any reconstruction of the distribution function, in the second part of the section we will give the details for the special case of piece-wise linear reconstructions.

The main idea behind such an extension of the original FKS method is to use a specific local piece-wise representation of the underlying data, for instance using polynomial reconstructions following classical MUSCL like technique [29] and to update the reconstructed distribution in the extremes points instead of updating it as usual in the cell centers. The starting point is, as before, the Discrete Velocity Model approximation of the original kinetic equation and a split into collision and transport terms. Only a first order splitting is considered in this work and the extension to higher order splitting is under study.

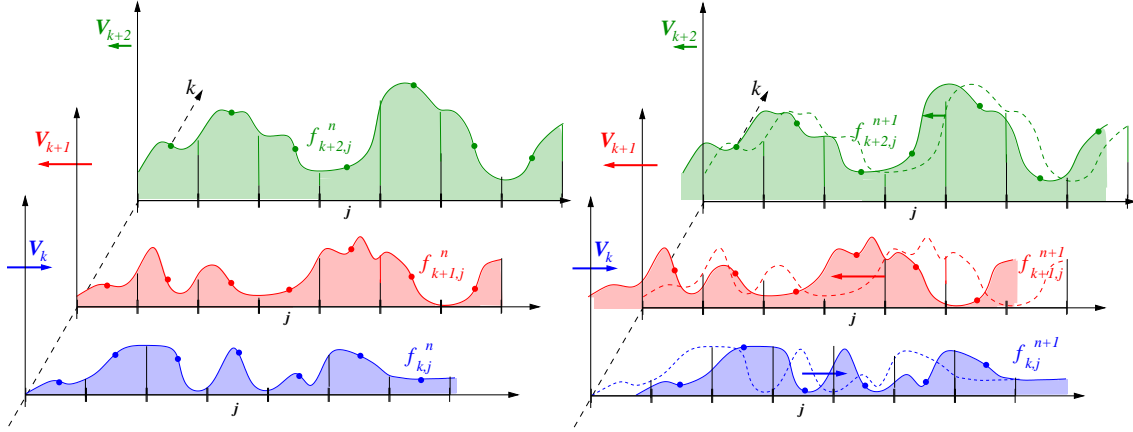


Figure 2: Illustration of the transport scheme in 1D for the high order FKS scheme. Left panel before transport step, right panel after transport step. Each discrete velocity (index k) drives its own transport equation with velocity v_k . The representation of f is made by means of continuous function. The shape of the entire function has not changed during the transport but the cell-centered values (bullets) may have.

Initial data representation. Let, as before, $f_{j,k}^0$ be the point-wise values at initial time t^0 of the distribution f , $f_{j,k}^0 = f(x_j, v_k, t^0)$ at cell center x_j for the lattice velocity v_k . Now, we define a space piece-wise polynomial reconstruction \tilde{f}_k^0 of a given order p starting from point-wise values $f_{j,k}^0$ for each velocity of the lattice v_k in cell center of Ω_j . Then, as in the first order case starting from data \tilde{f}_k^0 at time index 0, the exact solution of (12) is simply

$$f_k^{*,1}(x) = \tilde{f}_k^0(x - v_k \Delta t), \quad \forall x \in \Omega. \quad (21)$$

In other words, the entire piece-wise polynomial function defined by \tilde{f}_k^0 , is advected with velocity v_k during Δt .

Reconstruction in the collisionless case. If for one moment we simply consider the collisionless case, i.e. $Q(f) = 0 \forall f$, then the extension of this procedure to the generic time step index n yields

$$f_k^{*,n+1}(x) = \tilde{f}_k^n(x - v_k \Delta t), \quad \forall x \in \Omega, \quad (22)$$

where the key point is that the new transport step restarts from where the previous one was finished. As a consequence there is no re-projection of the distribution function f on the mesh points x_j to compute the new polynomial reconstruction as it is the case for classical semi-Lagrangian schemes. Therefore, in this pure convective case, this method is exact if the initial data are exactly represented by the initial polynomial reconstruction employed (the one at time 0), see figure 2 for a sketch of the transport phase in the one dimensional case when a continuous distribution function is considered.

Remark 1 In order to practically compute the solution of the transport step, one simply needs to store at the beginning of the computation for each point of the mesh a vector of size $p + 1$

where p is the order of the polynomial reconstruction containing the coefficients of the polynomial. Successively, in order to know the value of the distribution f everywhere, for each velocity of the lattice one needs to keep track of a single fixed point in the physical space (assuming that the transport velocity is constant and the mesh is uniform). This means that the computational cost associated to the solution of this step is very low compared to any kind of finite volume technique.

Reconstruction in the collisional case. In the case where collisions occur, their effect is to change the shape of the transported distribution in velocity space. Let $\tilde{f}_k^{*,n+1}(x)$ be the solution known for each point x in the physical space after the $(n+1)$ th transport step. The idea behind the construction of the schemes is to replace the classical solution computed in the center of the cell (first order forward Euler scheme has been employed for simplicity but no restriction on the time integrators is implied), that is

$$f_k^{n+1}(x_j) = \tilde{f}_k^{*,n+1}(x_j) + \Delta t Q \left(\tilde{f}_k^{*,n+1}(x_j) \right), \quad (23)$$

by

$$f_k^{n+1}(x_{\text{ext}_k}) = \tilde{f}_k^{*,n+1}(x_{\text{ext}_k}) + \Delta t \tilde{Q}_k^{n+1}(x_{\text{ext}_k}), \quad (24)$$

where x_{ext_k} are the extreme points of $\tilde{f}_k^{*,n+1}(x)$ which are in general different for each k . Let observe that there exists only M extremes points for each value k of the velocity mesh since the reconstruction is performed in such a way that the Discrete Maximum Principle is satisfied and then successively preserved since the transport phases keep the structure of the solution unchanged. This choice permits to minimize the loss of accuracy due to the numerical diffusion since in general the extreme points do not lie on the cell centers and thus they are lost at each time step. Notice also that the effect of (23) as well as of (24) is to compute the new value of the distribution in only one point x_j or alternatively on x_{ext_k} . This means that only the solution in the cell centers or in the extreme points have been updated with one of the two choices. However, the principle of the FKS method is to know the solution in each point of the domain in order to continue to the next time step. As a consequence in order to compute the next transport step and advance in time, we need to define a new continuous distribution function, that is for all point x . For this we employ the same initial polynomial reconstruction used for the distribution f at time 0 using these new M extrema as interpolation points. This ends one time step of the scheme. The only point that remains to specify is how, in practice, are computed the functions $\tilde{Q}_k^{n+1}(x_{\text{ext}_k})$? Three different possible choices as anticipated will be discussed in the next paragraph. They lead to three possible schemes. Only one choice is retained for the numerical simulation.

Computation of the collision operator in the extremes points. We identified three different possibilities in order to give an estimation of $\tilde{Q}_k^{n+1}(x_{\text{ext}_k})$ for the different lattice k and extreme points.

1. The first possibility is to observe that the distribution function is known everywhere after the transport step and so especially in x_{ext_k} . Thus one could simply get $\tilde{Q}_k^{n+1}(x_{\text{ext}_k})$ by direct computation from the values $\tilde{f}_k^{*,n+1}(x_{\text{ext}_k})$. However, this first simple and natural choice may be very expensive when realistic collision operators have to be used such as the Boltzmann operator (7). In fact, this choice implies the collision operator to be computed $N \times M$ times, i.e. the number of extreme points, at each time step. Unfortunately, this will be impracticable in multidimensional settings and with a Boltzmann like collision operator.
2. The second possibility is to compute at the M locations corresponding to the M cell centers the collision operator that is

$$Q \left(\tilde{f}_k^{*,n+1}(x_j) \right), \quad (25)$$

and then to interpolate the known values in the cell centers in order to get an estimation of $\tilde{Q}_k^{n+1}(x_{\text{ext}_k})$. This second choice will be largely computationally less expensive compared to the first proposed solution. However, the interpolation procedure will cause the same loss of accuracy that one gets with a classical semi-Lagrangian discretization since the information about the extreme points of the distribution function will not be used in this case to update the solutions; only the cell centers values will be used. The resulting scheme would perform very well in the non collisional or almost non collisional case since in this case extreme values are preserved (or almost preserved) by the transport phase. However this scheme would perform no better than a classical semi-Lagrangian scheme in the limit in which the number of collision becomes very large.

3. The third possibility is to compute the collision operator at the M locations corresponding to the M cell centers as before, i.e. $Q(\tilde{f}_k^{*,n+1}(x_j))$. Next we perform the same polynomial reconstruction employed for the distribution f for each velocity of the lattice v_k . This gives the function $\tilde{Q}_k^{n+1}(x)$, starting from the point-wise values $Q(\tilde{f}_k^{*,n+1}(x_j))$. The fundamental assumption for using such an approach lies on the fact that f , which gives the distribution of particles before the collision and $Q(f)$, which gives the distribution of particles after the collision, are supposed to evolve in a close relationship so that to share the same space topology. More precisely, our statement is that the collision operator modifies the values f_k for each velocity of the lattice, in fact the collisions redistribute the relative weights of the lattice velocity, but not their spacial distribution. This last solution, which will be the one implemented and tested, preserves the structure of the solution as in the first possibility but at a cost comparable to that of the second possibility. In fact only M evaluations of the collision operator are used in this case.

In order to detail the proposed approach (the third case) we need to define the polynomial reconstruction and an associated limiting procedure to ensure that new extrema are not generating spurious oscillations. In the following, for the sake of clarity, we choose a simple piece-wise linear reconstruction of the distribution function along with *ad hoc* limiters. Moreover we restrict ourselves to the simple case of BGK collisional operator. Note that the detailed construction of FKS schemes in multidimensions with higher order polynomial reconstructions for Boltzmann type of operator is the subject of a future work.

3.2 Linear polynomial reconstruction case

In this section we will focus on a FKS method which employs a polynomial reconstructions of degree one. The transport phase has been detailed in the previous paragraphs, it can however be summarized by the sketch in figure 3 for the specific case under study. The collisions are modeled by a relaxation towards the thermodynamics equilibrium $Q_{BGK} = \nu(M[f] - f)$. In this specific case, being the transport phase unaltered, the collision phase becomes on the grid points

$$f_k^{n+1}(x_j) = [\exp(-\nu\Delta t)] f_k^{*,n+1}(x_j) + [1 - \exp(-\nu\Delta t)] \mathcal{E}_k[U_j^{*,n+1}], \quad (26)$$

where $\mathcal{E}_k[U_j^{*,n+1}]$ is a consistent discretization of the Maxwellian distribution at the grid points. This consistent discretization only depends on the macroscopic moments $U(x_j, t^{*,n+1}) = U_j^{*,n+1}$ also defined on the grid points and satisfies the conservation of macroscopic quantities

$$U(x_j, t^{*,n+1}) = \sum_k \phi_k \mathcal{E}_k[U_j^{*,n+1}] \Delta v = \sum_k \phi_k f_k^{*,n+1}(x_j) \Delta v. \quad (27)$$

In the previous equation we have used the exact solution of equation $\partial_t f = Q_{BGK}$ in this case instead of the solution of the forward Euler scheme as before. Equation (26) furnishes the new

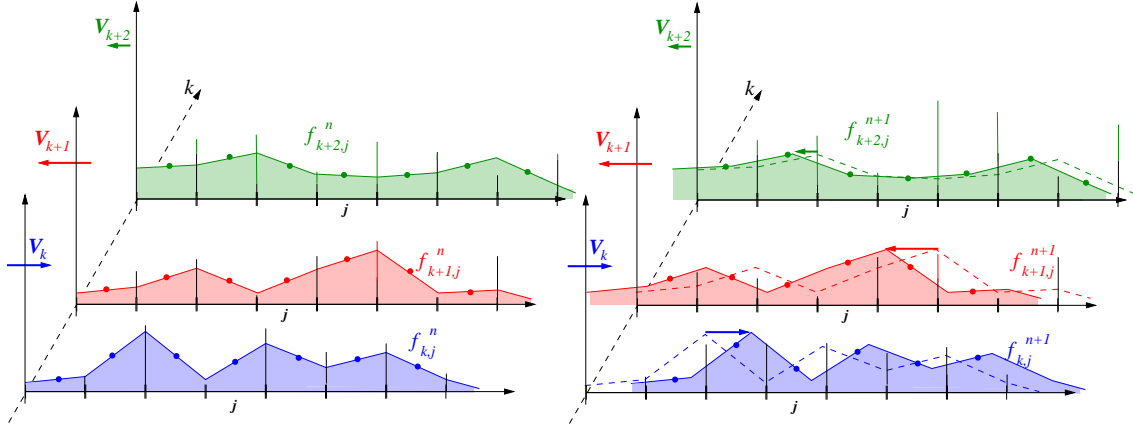


Figure 3: Illustration of the transport scheme in 1D for the high order FKS scheme with linear reconstruction. Left panel before transport step, right panel after transport step. Each discrete velocity (index k) drives its own transport equation with velocity v_k . The representation of f is made by means of continuous function. The shape of the entire function has not changed during the transport but the cell-centered values (bullets) may have.

value of the distribution f at time t^{n+1} only in the cell centers of the spacial cells for each velocity v_k , see the green bullets in figure 4. Then, since our aim is to update the distribution function at the extreme points x_{ext_k} , we need to evaluate the Maxwellian distribution in these points. In order to do that we use the same first order polynomial interpolation used for f but for the Maxwellian distribution, let observe the piece-wise linear green line in figure 4. These slopes are then transferred onto the Maxwellian distribution to create a continuous version of M from cell centered data (blue crosses in figure 4). However, this procedure causes loss of uniqueness in the definition of the Maxwellian state at the extreme points, in fact for each of the extreme points we get a left state $\mathcal{E}_k^-(x_{\text{ext}_k})$ and a right $\mathcal{E}_k^+(x_{\text{ext}_k})$ state, see figure 4. In order to restore uniqueness and to avoid the creation of extrema which may be possibly premises of spurious oscillations, the value of the Maxwellian function at these points is defined by taking the average between the left and the right states when the slopes on both sides have the same sign. This average is weighted by the distance from the cell center. On the other hand, when the slopes of the left and right states are of opposite signs, then the value of the equilibrium function is fixed as the minimum between the left and the right states if the left slope is positive (presence of a maximum), or as the maximum if the left slope is negative (presence of a minimum). For all cases, this value is denoted by $\mathcal{E}_k[U_j^{*,n+1}](x_{\text{ext}_k})$. Finally the new distribution f_k^{n+1} at any extreme point x_{ext_k} is defined by summing the values of the transported distribution and of the reconstructed Maxwellian distribution at the M points at which the transported distribution function reaches its extrema, that is

$$f_k^{n+1}(x_{\text{ext}_k}) = [\exp(-\nu\Delta t)] f_k^{*,n+1}(x_{\text{ext}_k}) + [1 - \exp(-\nu\Delta t)] \mathcal{E}[U^{n+1}]_k(x_{\text{ext}_k}). \quad (28)$$

From the knowledge of these new extreme values, we redefine the new reconstructed distribution f by the same first order polynomial interpolation employed at time t^n using the points x_{ext_k} as interpolation nodes. Consequently we are ready to compute another transport phase and this ends one time step of the scheme.

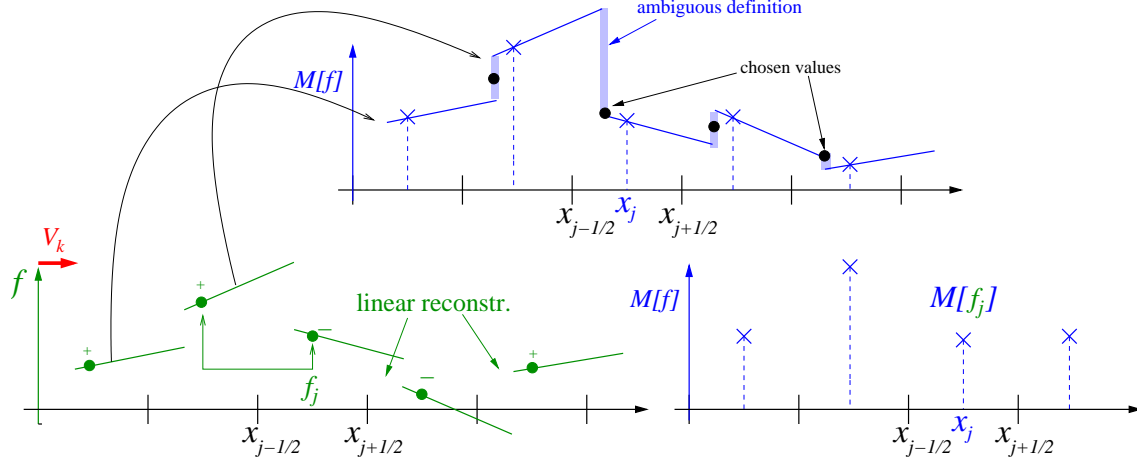


Figure 4: Illustration of the reconstruction technique employed in the R-FKS — From the distribution function data one reconstructs an upwind piece-wise linear polynomial (green on the left panel). From the distribution functions known at the cell centers we evaluate the Maxwellian also at cell centers (blue cross). Last on the top panel, we employ the same slope from the distribution function which passes through the Maxwellian cell center value. An ambiguous definition at each node is clarified by choosing the less extreme values in case the slopes are of different signs and the mean value otherwise, see the black bullets.

4 Numerical experiments

In this section we test the FKS scheme with piece-wise linear reconstructions against the original FKS schemes (referred to as R-FKS and FKS respectively) and some classical semi-Lagrangian (SL-) schemes of first order (referred to as 'SL-Upwind') and second order of accuracy (referred to as 'SL-MUSCL'). The SL-MUSCL scheme employs a simple piece-wise linear reconstruction with van Leer flux limiter [29]. Note that most of semi-Lagrangian second-order limited schemes would produce rather equivalent results.

The methodology of verification is revolving around three test cases:

- Numerical convergence test. Given a smooth solution for different value of ν , we observe that the numerical method is nominally second order accurate.
- Riemann problem using Sod like initial data
 - for $\nu = 10^4$ which consists of an almost pure hydrodynamics situation where genuine shock wave and discontinuous solutions may appear;
 - for $\nu = 10^3$ which is a transition regime where discontinuities are no more present but steep fronts are still present;
 - for $\nu = 10^2$ which is the entrance door towards the full collisional regime for which physical diffusion dominates any other non-linear waves.
- Highly oscillating initial data which are generating a large number of discontinuities even in the kinetic regime $\nu = 10^{-2}$. This problem requires a low dissipative scheme to maintain accurate results. For this test we will present mesh convergence and efficiency studies.

One expects, and we will show that it is indeed the case, that the new FKS scheme with reconstruction (R-FKS) is more accurate than FKS and the SL-Upwind schemes, and has the same -or possibly better- accuracy than SL-MUSCL scheme. At the same time one expects that the R-FKS scheme be computationally less expensive than the SL-MUSCL scheme. In an ideal situation the cost of R-FKS should be close to the FKS scheme, which is already known to be an inexpensive scheme compared to classical SL strategy. Finally, as expected, we will show that for a fixed error the R-FKS demands less computational resources than SL-MUSCL, FKS and SL-Upwind in 1D×1D assuming that, if so, this situation will be vastly more interesting in 2D×2D and 3D×3D computations. In all simulations we set the CFL number to one for all schemes.

4.1 Convergence test problem

This problem is initialized with a smooth macroscopic density, velocity and temperature data

$$\rho(x, t = 0) = 1 + \frac{1}{2} \sin(2\pi x), \quad u(x, t = 0) = 0, \quad T(x, t = 0) = 5 + \frac{1}{2} \sin(2\pi x). \quad (29)$$

where $\Omega = [0, 1]$ and an initial thermodynamics equilibrium is chosen, i.e. $f(x, v, t = 0) = M[f](x, v, t = 0)$. The final time is set to $t_{\text{final}} = 0.025$. The initial truncation of the velocity space is $L_v = [-15, 15]$. The meshes are uniform both in velocity and physical spaces with $N = 50$ and $M_j = 100j$ cells with $j = 1, 2, 4, 8, 16, 32$ and 64 respectively.

In figure 5 are presented the initial data (top panel) and the final solution for $\nu = 10^1, 10^2, 10^4$ from left to right. Density, velocity and temperature are displayed when $M_1 = 100$ cells are considered. According to our initialization the cell centers for mesh M_1 are also cell centers for any mesh M_i ,

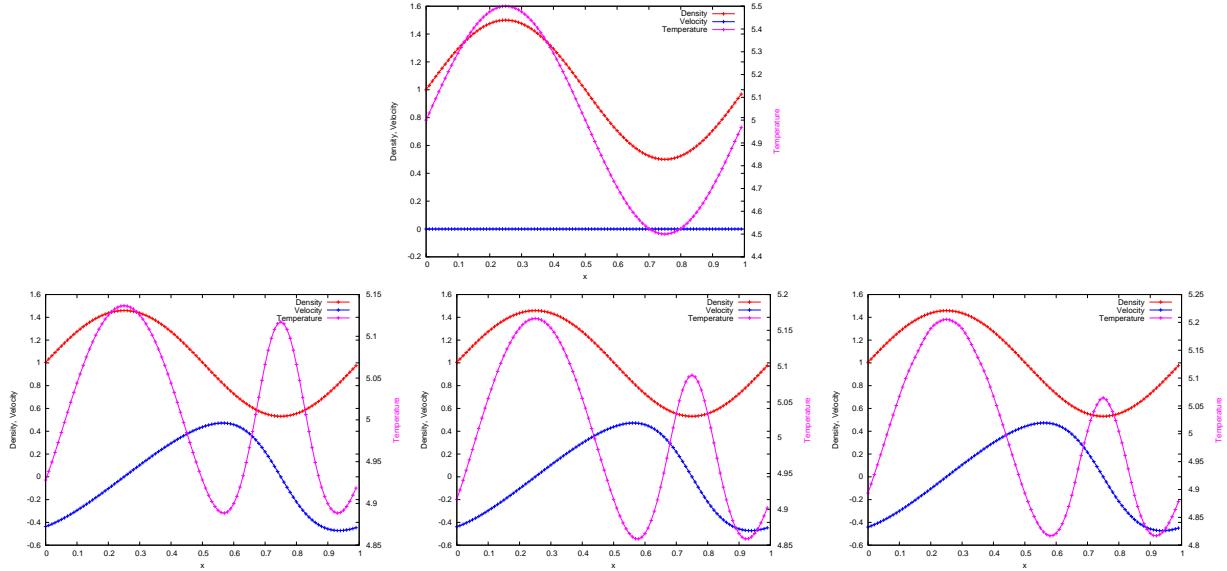


Figure 5: Convergence test case results for $M_1 = 100$ cells — Top panel: initial data — Left to right panels: final solution for $\nu = 10^1, 10^2, 10^4$.

$1 \leq i \leq 64$. Consequently we can compute an estimation of the numerical order of convergence

Meshes M_j, M_{j+1}, M_{j+2}	Order of convergence		
	$\nu = 10^1$	$\nu = 10^2$	$\nu = 10^4$
1, 2, 4	1.010	1.056	1.776
2, 4, 8	9.404	6.695	4.569
4, 8, 16	2.711	2.012	2.584
8, 16, 32	2.658	2.043	2.556
16, 32, 63	1.968	1.948	2.357

Table 1: Convergence test case for a smooth solution computed from three successively refined meshes M_j, M_{j+1}, M_{j+2} via equation (30) and for three collisional frequencies.

using three consecutive meshes of indices $j, j+1, j+2$ as

$$p_j = \log_2 \left(\frac{\sum_{i=1}^{M_1} |\rho_i^j - \rho_{2i}^{j+1}|}{\sum_{i=1}^{M_1} |\rho_{2i}^{j+1} - \rho_{4i}^{j+2}|} \right), \quad j = 1, 2, 4, 8, 16. \quad (30)$$

In order to properly measure the error due to the space discretization, we make the time discretization error very small by fixing the time step Δt to the value needed for the finest mesh, that is $\Delta t = (\Delta x)_{64} / \max(|v|) \simeq 10^{-5}$, where $(\Delta x)_{64} = 1/6400$. Moreover, in order to observe the convergence rate for different regimes, we consider three different collision frequencies : $\nu = 10^1$, 10^2 and 10^4 . The results are gathered in table 1 and we observe that, for the three collisional frequencies considered, the nominal orders of accuracy are approaching 2. The first two orders of accuracy that can be computed present some erratic behaviors because the numerical solution is not yet mesh-converged.

4.2 Riemann problem using Sod like initial data

In this paragraph we study the classical Riemann problem with Sod like initial data

$$\begin{aligned} \int_{\mathbb{R}} f dv &= 1, \quad \int_{\mathbb{R}} f v dv = 0, \quad \int_{\mathbb{R}} f v^2 dv = 2.5, & \text{if } x \leq L/2, \\ \int_{\mathbb{R}} f dv &= 0.125, \quad \int_{\mathbb{R}} f v dv = 0, \quad \int_{\mathbb{R}} f v^2 dv = 0.25, & \text{if } x > L/2, \end{aligned}$$

with $\Omega = [0, 1]$ the domain length and initial thermodynamics equilibrium, i.e. $f(x, v, t = 0) = M[f](x, v, t = 0)$. The CFL condition is fixed for all different tests to $\Delta t = \Delta x / \max(|v|)$, where $\max(|v|)$ corresponds to the maximum speed of the particles which is fixed by the initial truncation of the velocity space $L_v = [-20, 20]$. The same initial data are used to run three different problems corresponding to three different collision frequencies : $\nu = 10^2$, 10^3 and 10^4 . The meshes are uniform both in velocity and in the physical space and the number of points is chosen respectively as $N = 50$ and $M = 300$.

In figure 6 is reported the density of the computed solution for a final time $t_{\text{final}} = 0.07$, in figure 7 the velocity and in figure 8 the temperature for the three different collision frequencies $\nu = 10^4$ (top panels), 10^3 (middle panels), 10^2 (bottom panels). On the left side the solution on the entire domain is plotted while on the right side a magnification of the solutions in some interesting locations is reported. In particular for the density the zoom is around the region in which the

solution develops a contact discontinuity in the fluid dynamic limit, for the velocity around the region in which the solution forms a shock wave in the limit, while for the temperature the zooms are around the region of the contact discontinuity for $\nu = 10^4$ and around the region of the shock wave for $\nu = 10^3$ and $\nu = 10^2$. The results obtained by the four schemes are reported on each plot: FKS in red, R-FKS in blue, SL-Upwind in magenta and SL-MUSCL in green.

The results show that SL-MUSCL and R-FKS are clearly more accurate than SL-Upwind and FKS in the highly collisional case $\nu = 10^4$. In particular, for the density we observe that the contact discontinuity is better captured by the R-FKS scheme compared to the SL-MUSCL, while for the temperature and the velocity the solutions are almost superimposed. The same behavior is observed for the shock wave in the fluid dynamic limit, the solution is better captured (less points on the shock wave are present) by the R-FKS scheme in comparison with the three other schemes employed, while the complexity of the scheme is the one of a first order scheme (only first order polynomial are employed). On the other hand, when ν decreases, the differences between the schemes are less pronounced as the physical diffusion takes over on the waves. However, for $\nu = 10^3$ we still observe a difference between the four solutions, in particular the two semi-Lagrangian schemes SL-Upwind and SL-MUSCL are more diffusive than the two FKS schemes (a convergence test, not reported, shows that the position of the waves converges to the one of the FKS method when the number of points increases). In addition, comparing FKS and R-FKS we observe that R-FKS produces slightly less diffusive results. Finally, in the case in which the kinetic effects are large, i.e. $\nu = 10^2$ the four solutions are almost superimposed and it is difficult to appreciate the differences between the methods, only the SL-Upwind is clearly slightly more diffusive than the others.

In order to better appreciate the behavior of our R-FKS method in the next paragraph we consider a case test in which in the kinetic regime many oscillations are present, this problem permits to enhance specific characteristics of the schemes.

4.3 Highly oscillating kinetic problem

The previous test case was clear in illustrating the gain of the R-FKS scheme compared to the other schemes close to the fluid limit and for intermediate regimes, i.e. $\nu = 10^3$, while the gain was less obvious when the regime was far from the thermodynamics equilibrium. Contrarily, the test case in this section is designed to illustrate the anti-diffusive behavior of the FKS methods compared to classical semi-Lagrangian schemes in a kinetic regime. In order to do that, let us consider an homogeneous gas, $\rho = 1$, at constant temperature, $T = 5$ and at rest $u = 0$ on a domain $\Omega = [0, 1]$. This gas is further animated by a stair case oscillating velocity field ($u = \pm 1$) over a period $0 < \delta \ll L$ as

$$u(x) = \begin{cases} +1 & \text{if } x \in [x_{2k} - \delta/2 : x_{2k} + \delta/2] \cap [0.25; 0.75], \\ -1 & \text{if } x \in [x_{2k+1} - \delta/2 : x_{2k+1} + \delta/2] \cap [0.25; 0.75], \\ 0 & \text{else} \end{cases} \quad \forall 0 \leq k \leq M/2, \quad (31)$$

where $\{x_i\}_{i=1,M}$ is the spacial mesh, see figure 9 for an illustration of the initial velocity field.

We then consider a kinetic regime, $\nu = 10^2$, for which kinetic effects are non negligible. In this situation, the highly oscillating velocity field generates waves emanating from each discontinuity. These waves later interact, creating secondary waves further in interaction and so on. The physical relaxation will damp some of these waves, their amplitude and their structure, and, any low dissipative numerical method must reproduce such behaviors. Associated with this physical dissipation, due to the embedded numerical diffusion of the schemes, some small structures may also be lost during the time evolution, or, at least, they may be damped. The goal of this test is to show that the class of FKS methods are capable to maintain in time these small structures whereas

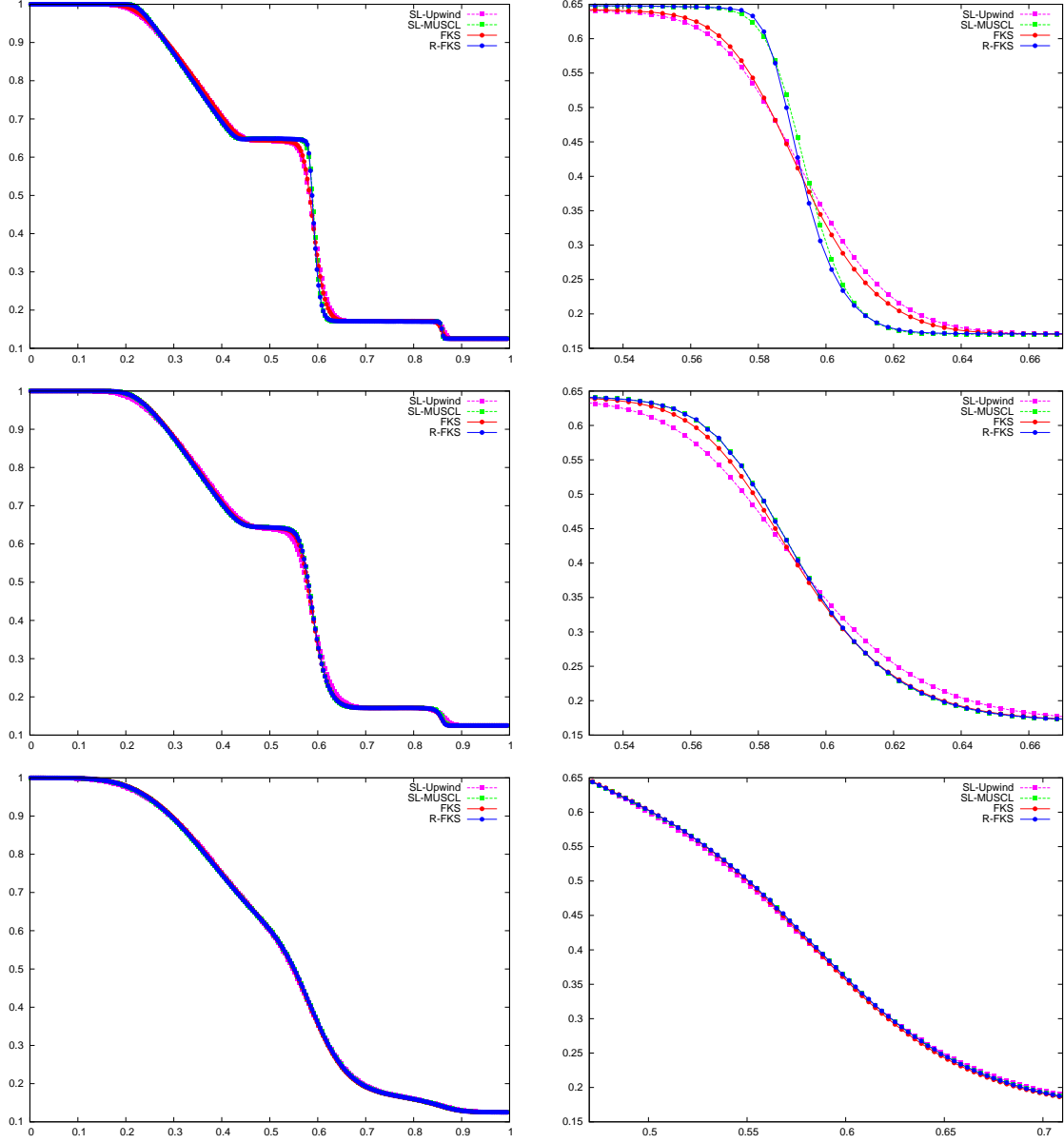


Figure 6: Riemann problem — Solution at $t_{\text{final}} = 0.07$ for the density. Left full solution, right zoom close to the region in which the solution develops a contact discontinuity in the limit of infinite collisions. Collision frequency $\nu = 10^4$ (top), 10^3 (middle), 10^2 (bottom).

they are, for a large part, lost using a semi-Lagrangian approach due to the intrinsic numerical diffusion.

In the following we consider two kind of tests which measure the performances of the schemes. The first one is a mesh convergence test while the second one is an efficiency test.

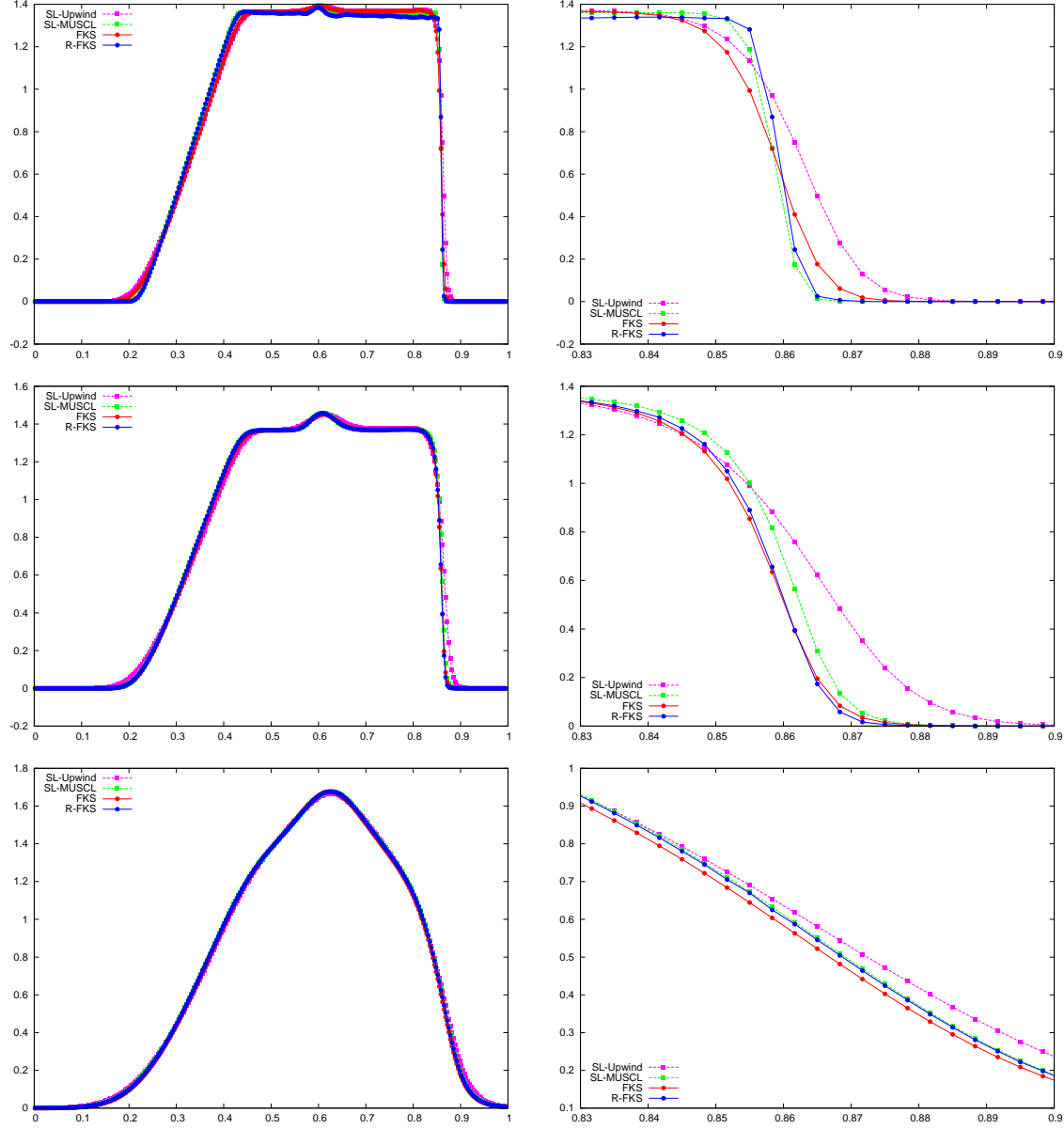


Figure 7: Riemann problem — Solution at $t_{\text{final}} = 0.07$ for the velocity. Left full solution, right zoom close to the region in which the solution develops a shock wave in the limit of infinite collisions. Collision frequency $\nu = 10^4$ (top), 10^3 (middle), 10^2 (bottom).

4.3.1 Mesh convergence study

For the first set of simulations, the final time is fixed at $t_{\text{final}} = 0.025$, while we choose $\delta = 0.02$. Then, an increasing sequence of spacial meshes is used for the four schemes: $M = 600, 1200, 2400$ and 4800 , while both the number of velocity cells $N = 50$ and the velocity domain $L_v = [-30 : 30]$

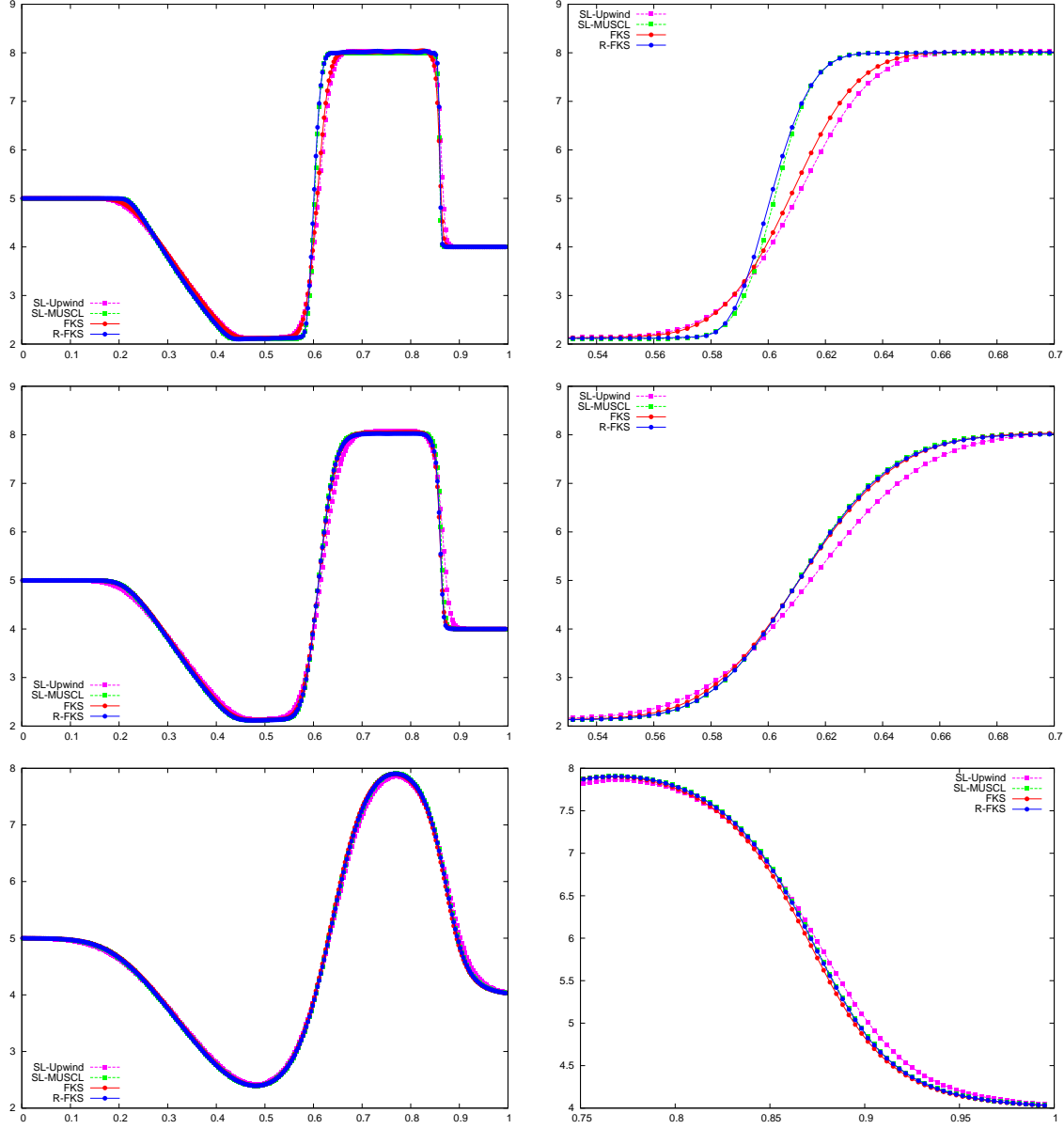


Figure 8: Riemann problem — Solution at $t_{\text{final}} = 0.07$ for the temperature. Left full solution, right zoom close to the region in which the solution develops a contact discontinuity (top) or a shock wave (middle and bottom) in the limit of infinite collisions. Collision frequency $\nu = 10^4$ (top), 10^3 (middle), 10^2 (bottom).

are kept fixed. In figure 10 we present the density, velocity and temperature for the four schemes when 600 cells are used. Observing this figure the schemes seem to capture the same main waves and they seem to overlap apart from the R-FKS which seems to oscillate. But a closer view shows that this is not the case. Thus, in order to enhance the differences between the schemes and

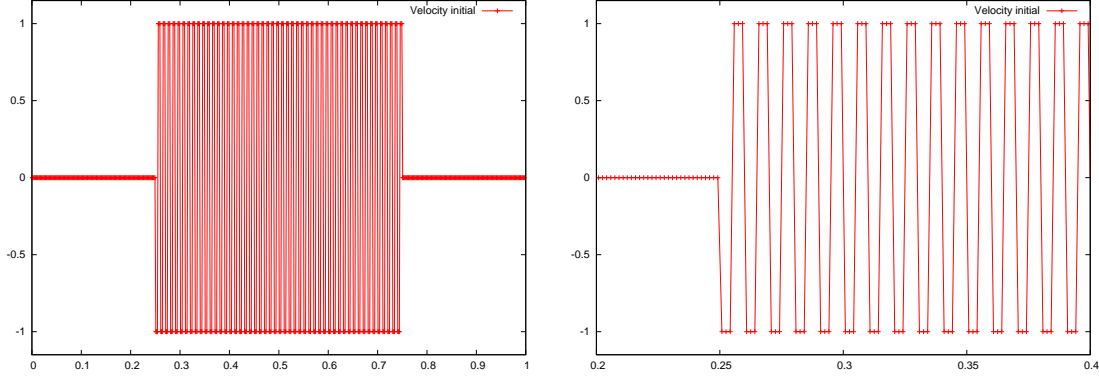


Figure 9: Initial velocity field for the highly oscillating kinetic test case — $\delta = 0.02$ and 600 mesh cells are used — Left: full view on Ω , right: zoom on $[0.2, 0.4]$.

investigate the small scale structures of the solution, in the next figures, we present magnifications on particular locations for each of the macroscopic variables. In figure 11 we present the mesh convergence study for the density variable with four different meshes: 600, 1200, 2400 and 4800 cells. Velocity and temperature variable magnifications results are displayed in figure 12 and 13 respectively. From these figures we observe that although the SL-Upwind scheme can capture the main flow, most of the small scale structures are damped out due to an excessive numerical diffusion. The same but less pronounced effect can be observed for the SL-MUSCL scheme. Indeed its embedded limiter¹ has a known tendency to clip extrema, which in this extreme case yields to excessive numerical diffusion. When the mesh is refined the SL-Upwind scheme starts to develop a wavy behavior which is a remembrance of the true physical oscillations. SL-MUSCL scheme, being second-order accurate is capturing more oscillations, but, is nonetheless still destroying the stair case initial profile. Remarkably the (R-)FKS scheme is able to maintain the small scale structures of the solution even with the coarser mesh. When the mesh is refined some more details of the flow are further captured. The spacial mesh refinement shows that SL-MUSCL, SL-Upwind and (R-)FKS converge towards the same solution. However, thanks to its almost exact transport phase, the FKS scheme family is drastically reducing the numerical diffusion in this kinetic regime. From those curves we can clearly see that the FKS family is genuinely able to maintain the structure of the solution for longer times and with much less mesh points whereas the intrinsic numerical diffusion of semi-Lagrangian schemes damps such structures.

4.3.2 Cost and efficiency study

Although the numerical results presented in this paper are obtained in the one dimensional case, it is nonetheless interesting to monitor the cost of each numerical scheme. Two situations are considered; the first one reports the CPU time at fixed meshes independent of the obtained accuracy², whereas the second situation measures the cost for a given accuracy (that is spacial mesh refinement is pursued up to the point the given error is attained). Because the exact solution does not exist for this problem we have to design a reference diagnostics to decide when a given scheme

¹Note that any other limiter will have more or less the same tendency because they are all designed to fulfill a Discrete Maximum Principle, and, as such it clips all t^{n+1} extrema to $\min/\max t^n$ data no matter if it represents a physically justified new extrema or if this extrema results from a numerical spurious oscillation.

²Basically it measures the time we have to wait when spacial/velocity meshes are chosen.

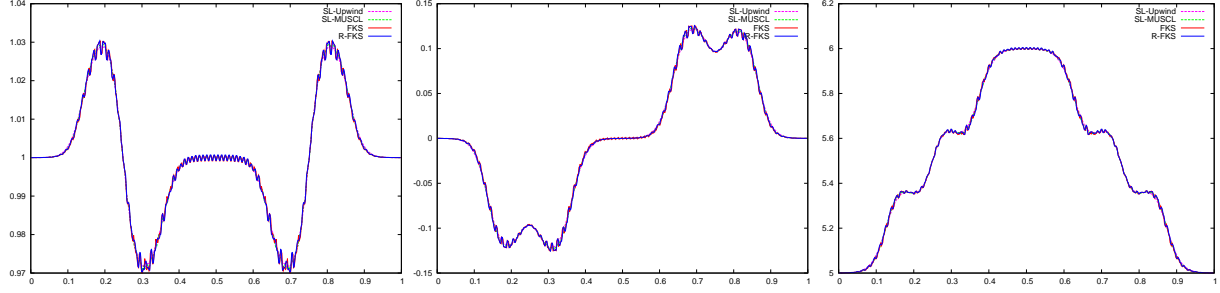


Figure 10: Oscillating problem — 600 cells — Solution at $t_{\text{final}} = 0.025$ for the density (left), mean velocity (middle) and temperature (right) with $\nu = 10^2$ and the SL-Upwind, SL-MUSCL, FKS and R-FKS schemes.

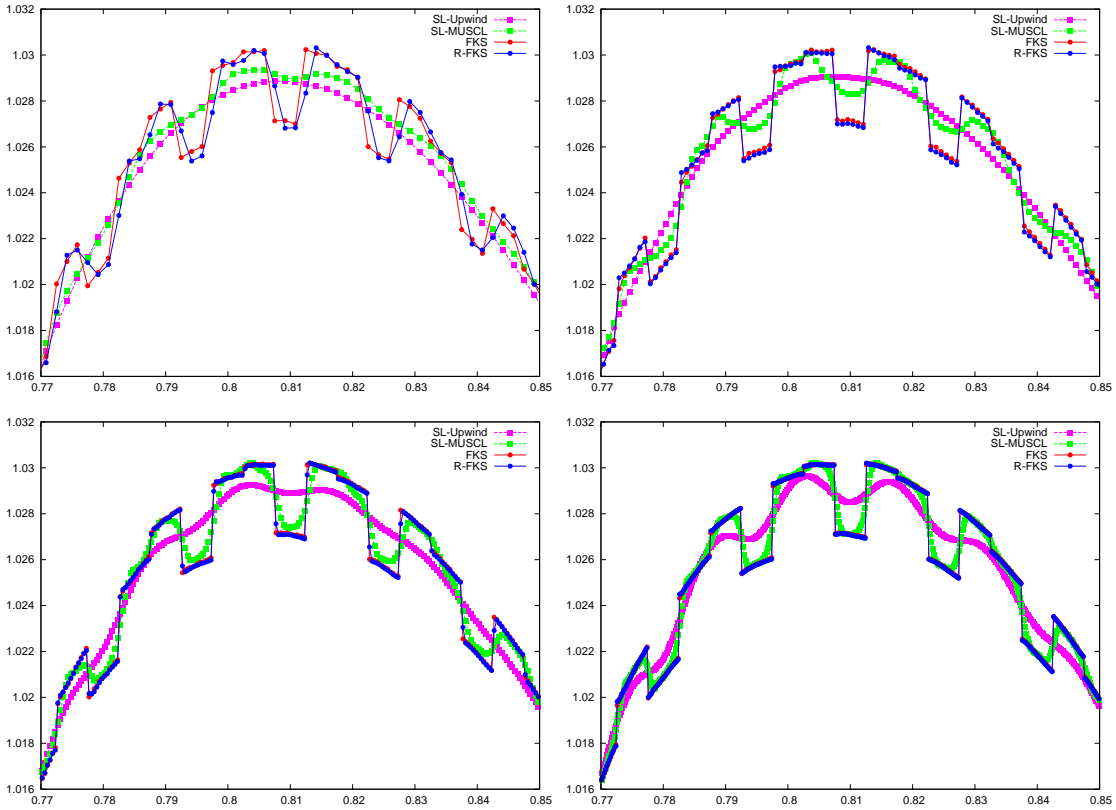


Figure 11: Oscillating problem — Zooms on density for 600, 1200, 2400 and 4800 cells (from top-left to bottom-right) — Solution at $t_{\text{final}} = 0.025$ with $\nu = 10^2$ for the SL-Upwind, SL-MUSCL, FKS and R-FKS schemes.

has attained an “acceptable” accuracy.

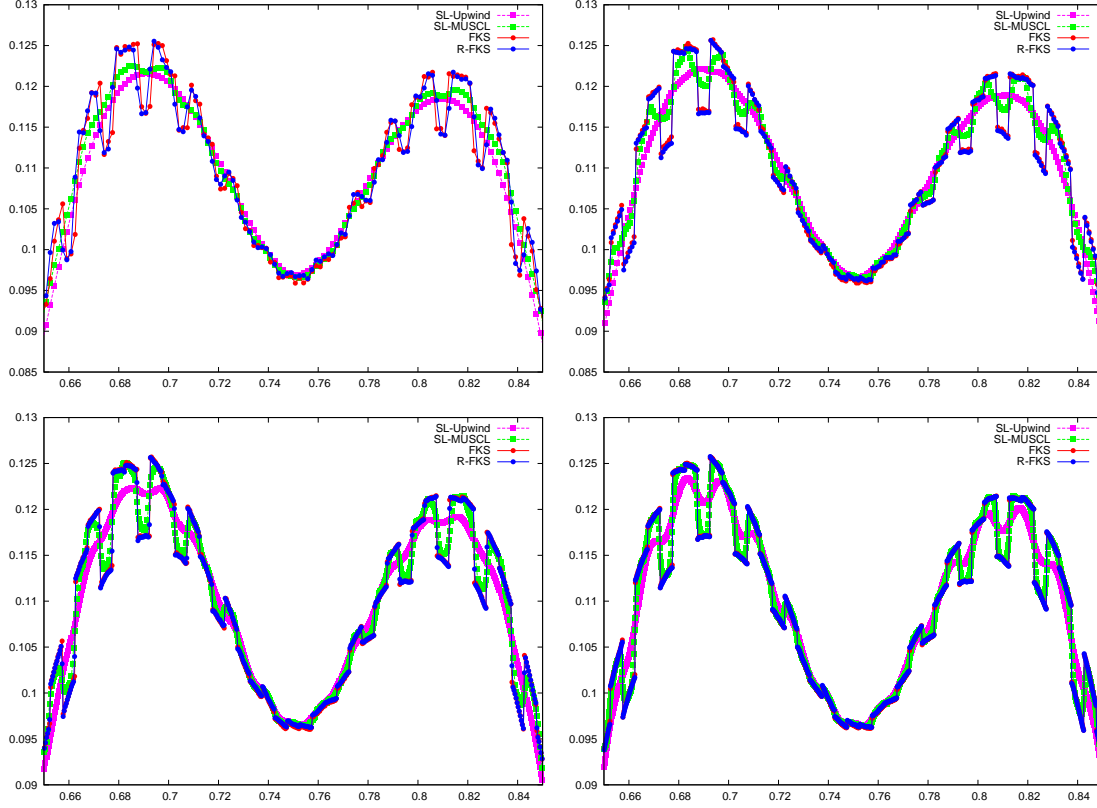


Figure 12: Oscillating problem — Zooms on velocity for 600, 1200, 2400 and 4800 cells (from top-left to bottom-right) — Solution at $t_{\text{final}} = 0.025$ with $\nu = 10^2$ for the SL-Upwind, SL-MUSCL, FKS and R-FKS schemes.

Cost at fixed meshes. We present in table 2 the cost of each of the previous simulations (from the mesh convergence study). For each simulation we report the number of time steps N_{cycle} and the CPU time T in second(s). We also compute the time per cycle by $T_{\text{cycle}} = T/N_{\text{cycle}}$ and time per cycle per physical cell N_c by $T_{\text{cell}} = T/N_{\text{cycle}}/N_c$. Last we propose the ratio of CPU time with respect to the corresponding SL-Upwind scheme results. From this table we can see that at fixed meshes FKS is faster than SL-Upwind and SL-MUSCL and R-FKS are about two times more expensive than FKS. When SL-MUSCL and R-FKS are compared then we can observe about 20% speed-up using R-FKS. Of course in a multidimensional setting the situation is completely different as we have shown in [18, 19], in this case the gain in term of computational cost is much larger for the FKS method. We expect the R-FKS method to be more expensive than FKS in more than one dimension, since we pass from a piecewise constant reconstruction to a linear reconstruction which leads to some additional operations but with a comparable computational cost [12].

Cost at fixed accuracy. However, as already mentioned, the previous table is not sufficient to give an idea of the performances of this class of methods. We also need to have a measure of the computational cost at fixed accuracy. For the oscillating test case we do not dispose of an exact reference solution. However, we have observed that both SL-MUSCL and the FKS family seem

Code.	N_v	Vel.	Cell # $N_c \times N_v$	Cycle N_{cycle}	Time T (s)	T/cycle T_{cycle} (s)	T/cell T_{cell} (s)	Time ratio vs SL-Upwind
SL-Upwind	50	[-15, 15]	600×50 $= 3 \times 10^4$	220	0.772 s	0.0035	5.86×10^{-6}	—
			1200×50 $= 6 \times 10^4$	441	3.317 s	0.0075	6.27×10^{-6}	—
			2400×50 $= 12 \times 10^4$	882	14.233 s	0.0161	6.72×10^{-6}	—
			4800×50 $= 24 \times 10^4$	1764	61.008 s	0.0346	7.21×10^{-6}	—
SL-MUSCL	50	[-15, 15]	600×50 $= 3 \times 10^4$	220	1.633 s	0.0074	1.24×10^{-5}	2.12
			1200×50 $= 6 \times 10^4$	441	6.784 s	0.0154	1.28×10^{-5}	2.05
			2400×50 $= 12 \times 10^4$	882	29.166 s	0.0331	1.38×10^{-5}	2.05
			4800×50 $= 24 \times 10^4$	1764	275.812 s	0.1564	3.26×10^{-5}	4.52
FKS	50	[-15, 15]	600×50 $= 3 \times 10^4$	220	0.518 s	0.0024	3.92×10^{-6}	0.67
			1200×50 $= 6 \times 10^4$	441	2.101 s	0.0048	3.97×10^{-6}	0.63
			2400×50 $= 12 \times 10^4$	882	8.754 s	0.0099	4.14×10^{-6}	0.62
			4800×50 $= 24 \times 10^4$	1764	38.544 s	0.0220	4.55×10^{-6}	0.63
R-FKS	50	[-15, 15]	600×50 $= 3 \times 10^4$	220	1.503 s	0.0068	1.14×10^{-5}	1.95
			1200×50 $= 6 \times 10^4$	441	5.981 s	0.0136	1.13×10^{-5}	1.80
			2400×50 $= 12 \times 10^4$	882	23.781 s	0.0270	1.12×10^{-5}	1.67
			4800×50 $= 24 \times 10^4$	1764	95.430 s	0.0541	1.13×10^{-5}	1.56

Table 2: Oscillating test case results for 600, 1200, 2400 and 4800 cells for the SL-Upwind, SL-MUSCL, FKS and R-FKS schemes. Time per cycle is obtained by $T_{\text{cycle}} = T/N_{\text{cycle}}$ and time per cycle per cell by $T_{\text{cell}} = T/N_{\text{cycle}}/N_c$.

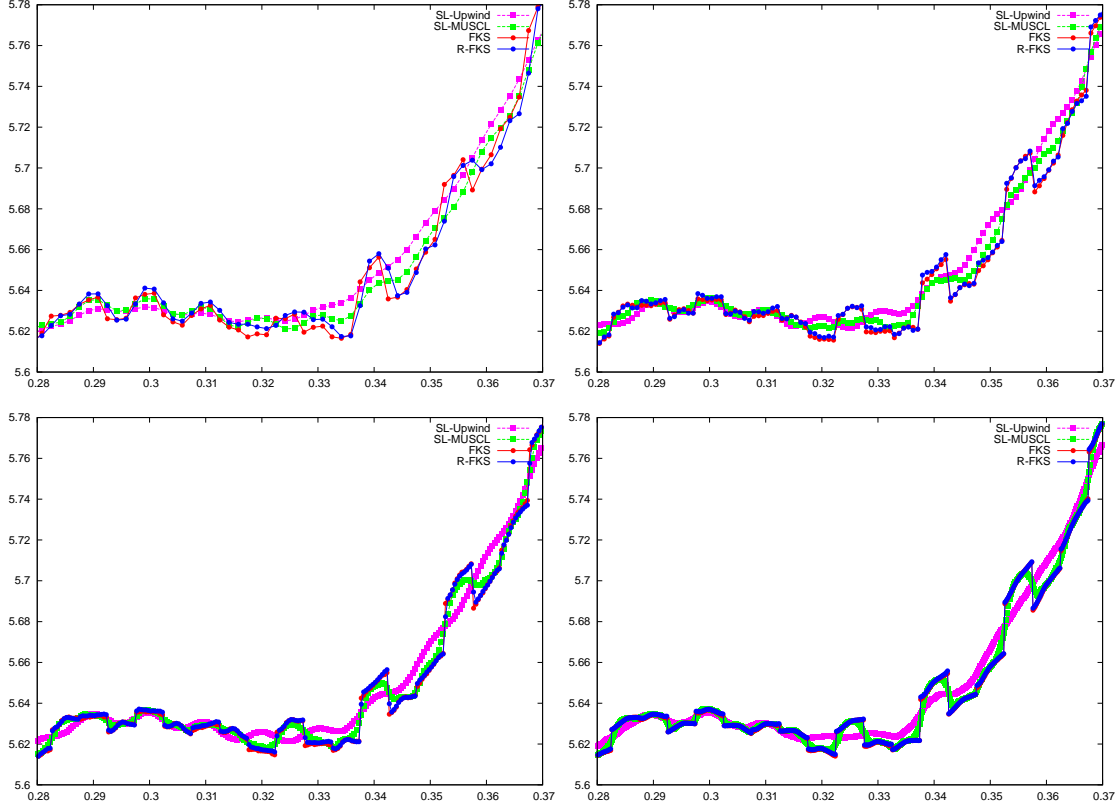


Figure 13: Oscillating problem — Zooms on temperature for 600, 1200, 2400 and 4800 cells (from top-left to bottom-right) — Solution at $t_{\text{final}} = 0.025$ with $\nu = 10^2$ for the SL-Upwind, SL-MUSCL, FKS and R-FKS schemes.

to capture an equivalent solution with discontinuities at the same locations when the mesh is fine enough. Moreover thanks to the mesh convergence study we estimated that the characteristics length between two discontinuities is of the order $\Delta l = 4.8 \times 10^{-3}$ (this corresponds to 23 cells of size $\Delta x = 1/4800$). In addition from the SL-MUSCL data we estimated that SL-MUSCL scheme spreads those discontinuities over $\mathcal{N} = 15$ cells. Therefore for the SL-MUSCL scheme it would require about $N_c \geq 2\mathcal{N}/\delta l$ cells to capture each of the “plateaus” between two discontinuities. In short SL-MUSCL scheme requires at minima $N_c = 6250$ cells to provide an accurate ‘enough’ numerical solution. For this amount of cells, according to table 2, the number of timesteps becomes $N_{\text{cycle}} = (0.3675) N_c$, indeed the the ratio between timesteps and number of cell is constant equal to $0.3675 = 1764/4800 = 882/2400 = 441/1200 (\simeq 220/600)$. Finally, given the time needed to update one cell during one timestep, T_{cell} from table 2, we can estimate the CPU time needed by the SL-MUSCL scheme via equation: $T = N_c \times N_{\text{cycle}} \times T_{\text{cell}}$, that is about $466\text{s} \simeq 7\text{mn}45\text{s}$. In table 3 we report these CPU times along with the other parameters from the four schemes. As expected from the numerical results, the numerical diffusion of SL-Upwind and SL-MUSCL schemes drastically penalizes their efficiency as they would need about 3 hours and 8 minutes respectively to compute a valid solution when the FKS family rounds about 2 to 3 seconds. We conclude by observing that, as usual, the computational cost analysis is a difficult task to accomplish and thus the numbers reported in the tables are only meant to provide a rough idea of the amount of

	SL-Upwind	SL-MUSCL	FKS	R-FKS
\mathcal{N}	150	15	2	2
δl	4.8×10^{-3}	4.8×10^{-3}	4.8×10^{-3}	4.8×10^{-3}
$N_c = 2\mathcal{N}/\delta l$	62500	6250	833	833
$N_{\text{cycle}} = (0.3675) N_c$	22969	2297	306	306
$T_{\text{cell}} \text{ (s)}$	7×10^{-6}	3.25×10^{-5}	4.55×10^{-6}	1.13×10^{-5}
$T = N_c \times N_{\text{cycle}} \times T_{\text{cell}}$	10048.83s (2h47mn)	466.55s (7mn45s)	1.16s	2.88s

Table 3: Oscillating test case results — Estimation of the CPU time T needed to capture the plateaus of estimated characteristics length δl . The schemes diffuse discontinuities over \mathcal{N} cells. The required number of cells is $N_c = 2\mathcal{N}/\delta l$, the number of time steps $N_{\text{cycle}} = (0.3675) N_c$, where $0.3675 = 1764/4800$ the ratio time steps/ N_c observed in table 2, $T = N_c \times N_{\text{cycle}} \times T_{\text{cell}}$.

resources that can already be saved in 1D by this family of schemes. Vastly more interesting gains are expected when 2D×2D and more important 3D×3D simulations will be performed.

5 Conclusion and perspectives

In this paper we have presented a new class of semi-Lagrangian methods based on high order polynomial reconstructions of the distribution function which extends the Fast Kinetic Scheme developed in [18, 19]. The main idea is based on updating the extreme points of the distribution function instead of updating the solution on the cells centers in order not to lose maxima and minima of the solution because of the numerical diffusion. In the first part of this paper, the method has been presented in all its generality and different approaches to compute the collision operator on the extrema are proposed. A critical analysis leads to choose the one which possesses the best properties in term of precision and efficiency. In a second part, details are then given for the specific case of a piece-wise linear polynomial reconstruction and a BGK collision operator. The technique is based on the assumption that the distribution function and its associated Maxwellian share the same space topology for equal velocity of the lattice. Finally, a min/max choices is made in order to treat the discontinuities of the Maxwellian function at the interface points.

In the third part of the paper, this new FKS scheme with reconstruction has been tested on two problems. The first one, a Riemann like problem, shows that in the fluid regime the method shares similar behaviors with respect a second-order accurate limited semi-Lagrangian (SL) scheme whereas the original FKS reproduces the results of a first-order accurate semi-Lagrangian scheme. For intermediate regimes, the new scheme is still better than the original FKS and the second order accurate semi-Lagrangian scheme but differences are extremely small. In a full kinetic regime this test case does not help to make a clear difference between first/second order accurate SL or FKS schemes. Conversely the second test case is a highly oscillatory kinetic test presenting many discontinuities which are damped by the classical semi-Lagrangian schemes. Contrarily the FKS family is able to maintain the discontinuities of the solution over time due to its natural anti-dissipation formulation. The results obtained by the FKS schemes are therefore superior in terms of precision and in terms of computational costs.

In the near future we will investigate the extension of this approach to the multi-dimensional case; the same reconstruction technique may apply but an analysis related to the choices to be done at interface surfaces/lines (3D/2D) should be carried out. We plan also to extend the new FKS scheme to the case of Boltzmann collisional operator.

Acknowledgements

This work was partly supported by Campus France under the PHC Gallileo 2015-2016 number 32272UL.

References

- [1] G.A.BIRD, *Molecular gas dynamics and direct simulation of gas flows*, Clarendon Press, Oxford (1994).
- [2] C.K. BIRSDALL, A.B. LANGDON, *Plasma Physics Via Computer Simulation*, Institute of Physics (IOP), Series in Plasma Physics (2004).
- [3] A.V. BOBYLEV, A. PALCZEWSKI, J. SCHNEIDER, *On approximation of the Boltzmann equation by discrete velocity models*. C. R. Acad. Sci. Paris Ser. I. Math. 320, (1995), pp. 639-644.
- [4] R. E. CAFLISCH, *Monte Carlo and Quasi-Monte Carlo Methods*, Acta Numerica (1998) pp. 1-49.
- [5] C. CERCIGNANI, *The Boltzmann Equation and Its Applications*, Springer-Verlag, New York, (1988).
- [6] L. CHACN, D. DEL-CASTILLO-NEGRETE, C.D. HAUCK, *An asymptotic-preserving semi-Lagrangian algorithm for the time-dependent anisotropic heat transport equation*, J. Comp. Phys. (2014), vol. 272, pp. 719-746.
- [7] C.Z. CHENG, G. KNORR, *The integration of the Vlasov equation in configuration space*, J. Comput. Phys., vol. 22 (1976), pp. 330-351.
- [8] N. CROUSEILLES, T. RESPAUD, E. SONNENDRUCKER, *A Forward semi-Lagrangian Method for the Numerical Solution of the Vlasov Equation*, Comp. Phys. Comm. 180, 10 (2009) 1730-1745.
- [9] N. CROUSEILLES, M. MEHRENBARGER, E. SONNENDRUCKER, *Conservative semi-Lagrangian schemes for Vlasov equations*, J. Comp. Phys., (2010) pp. 1927-1953.
- [10] P. DEGOND, G. DIMARCO, L. PARESCHI, *The Moment Guided Monte Carlo Method*, Int. J. Num. Meth. Fluids, Vol.67, (2011), pp. 189-213.
- [11] L. DESVILLETES, S. MISCHLER, *About the splitting algorithm for Boltzmann and BGK equations*. Math. Mod. & Meth. in App. Sci. bf 6, (1996), pp. 1079-1101.
- [12] G. DIMARCO, C. HAUCK, R. LOUBÈRE, *Multidimensional high order FKS schemes for the Boltzmann equation*, in progress.
- [13] G. DIMARCO, L. PARESCHI, *Numerical methods for kinetic equations*, ACTA Numerica, Vol. 23 (2014), pp. 369-520.
- [14] G. DIMARCO, L. PARESCHI, *Asymptotic preserving implicit-explicit Runge-Kutta methods for non linear kinetic equations*, SIAM J. Num. Anal., Vol. 49, (2011), pp. 2057-2077 ..
- [15] G. DIMARCO, L. PARESCHI, *Exponential Runge-Kutta methods for stiff kinetic equations*, SIAM J. Num. Anal., Vol. 51, (2013), pp. 1064-1087.
- [16] G. DIMARCO, L. PARESCHI, *A Fluid Solver Independent Hybrid method for Multiscale Kinetic Equations*, SIAM J. Sci. Comput. Vol. 32, (2010), pp. 603-634.
- [17] G. DIMARCO, *The hybrid moment guided Monte Carlo method for the Boltzmann equation*. Kin. Rel. Models, Vol 6, pp. 291-315 (2013).

- [18] G. DIMARCO, R. LOUBÈRE, *Towards an ultra efficient kinetic scheme. Part I: basics on the BGK equation*, J. Comput. Phys., Vol. 255, 2013, pp 680-698.
- [19] G. DIMARCO, R. LOUBÈRE, *Towards an ultra efficient kinetic scheme. Part II: the high order case*, J. Comput. Phys., Vol. 255, 2013, pp 699-719.
- [20] F. FILBET, G. RUSSO, *High order numerical methods for the space non-homogeneous Boltzmann equation*. J. Comput. Phys., 186 (2003), 457–480.
- [21] F. FILBET, E. SONNENDRÜCKER, P. BERTRAND, *Conservative Numerical schemes for the Vlasov equation*. J. Comput. Phys. 172, (2001) pp. 166–187.
- [22] I.M. GAMBA, S. H. THARKABHUSHAMAN, *Spectral - Lagrangian based methods applied to computation of Non - Equilibrium Statistical States*. J. Comput. Phys. 228, (2009) pp. 2012–2036.
- [23] E.P. GROSS P.L. BATHNAGAR, M. KROOK, *A model for collision processes in gases. I. small amplitude processes in charged and neutral one-component systems* Phys. Rev. 94 (1954), pp. 511–525.
- [24] Y. GÜÇLÜ, W.N.G. HITCHON, *A high order cell-centered semi-Lagrangian scheme for multi-dimensional kinetic simulations of neutral gas flows*. J. Comput. Phys. 231, (2012) pp. 3289-3316.
- [25] Y. GÜÇLÜ, A.J. CHRISTLIEB, W.N.G. HITCHON, *Arbitrarily high order Convected Scheme solution of the Vlasov-Poisson system* J. Comput. Phys. 270, (2014) pp. 711-752.
- [26] T. HOMOLLE, N. HADJICONSTANTINO, *A low-variance deviational simulation Monte Carlo for the Boltzmann equation*. J. Comput. Phys., Vol 226 (2007), pp 2341–2358.
- [27] T. HOMOLLE, N. HADJICONSTANTINO, *Low-variance deviational simulation Monte Carlo*. Phys. Fluids, Vol 19 (2007), 041701.
- [28] S. JIN, *Efficient asymptotic-preserving (ap) schemes for some multiscale kinetic equations*, SIAM J. Sci. Comput. Vol. 21 (1999), 441454.
- [29] R. J. LEVEQUE, *Numerical Methods for Conservation Laws*, Lectures in Mathematics, Birkhauser Verlag, Basel (1992).
- [30] L. MIEUSSENS, *Discrete Velocity Model and Implicit Scheme for the BGK Equation of Rarefied Gas Dynamic*, Math. Models Meth. App. Sci., Vol. 10, (2000), 1121–1149.
- [31] A. PALCZEWSKI, J. SCHNEIDER, A.V. BOBYLEV, *A consistency result for a discrete-velocity model of the Boltzmann equation*. SIAM J. Numer. Anal. 34, (1997) pp. 1865-1883.
- [32] A. PALCZEWSKI, J. SCHNEIDER, *Existence, stability, and convergence of solutions of discrete velocity models to the Boltzmann equation*. J. Statist. Phys. 91, (1998) pp. 307-326.
- [33] L. PARESCHI, G. RUSSO, *Numerical solution of the Boltzmann equation I: Spectrally accurate approximation of the collision operator*, SIAM J. Numer. Anal. Vol. 37, pp. 12171245 (2000).
- [34] L. PARESCHI, G. RUSSO AND G. TOSCANI, *Fast spectral methods for the Fokker PlanckLandau collision operator*, J. Comput. Phys. Vol 165, pp. 216236 (2000).
- [35] L. Pareschi, G. Toscani, *Interacting multi-agent systems. Kinetic equations and Monte Carlo methods*, Oxford University Press, USA, (2013).
- [36] M. SHOUCRI, G. KNORR, *Numerical integration of the Vlasov equation*. J. Comput. Phys., vol. 14 (1974), pp. 8492.
- [37] G. STRANG, *On the construction and the comparison of difference schemes*. SIAM J. Numer. Anal., (1968), pp. 506–517.